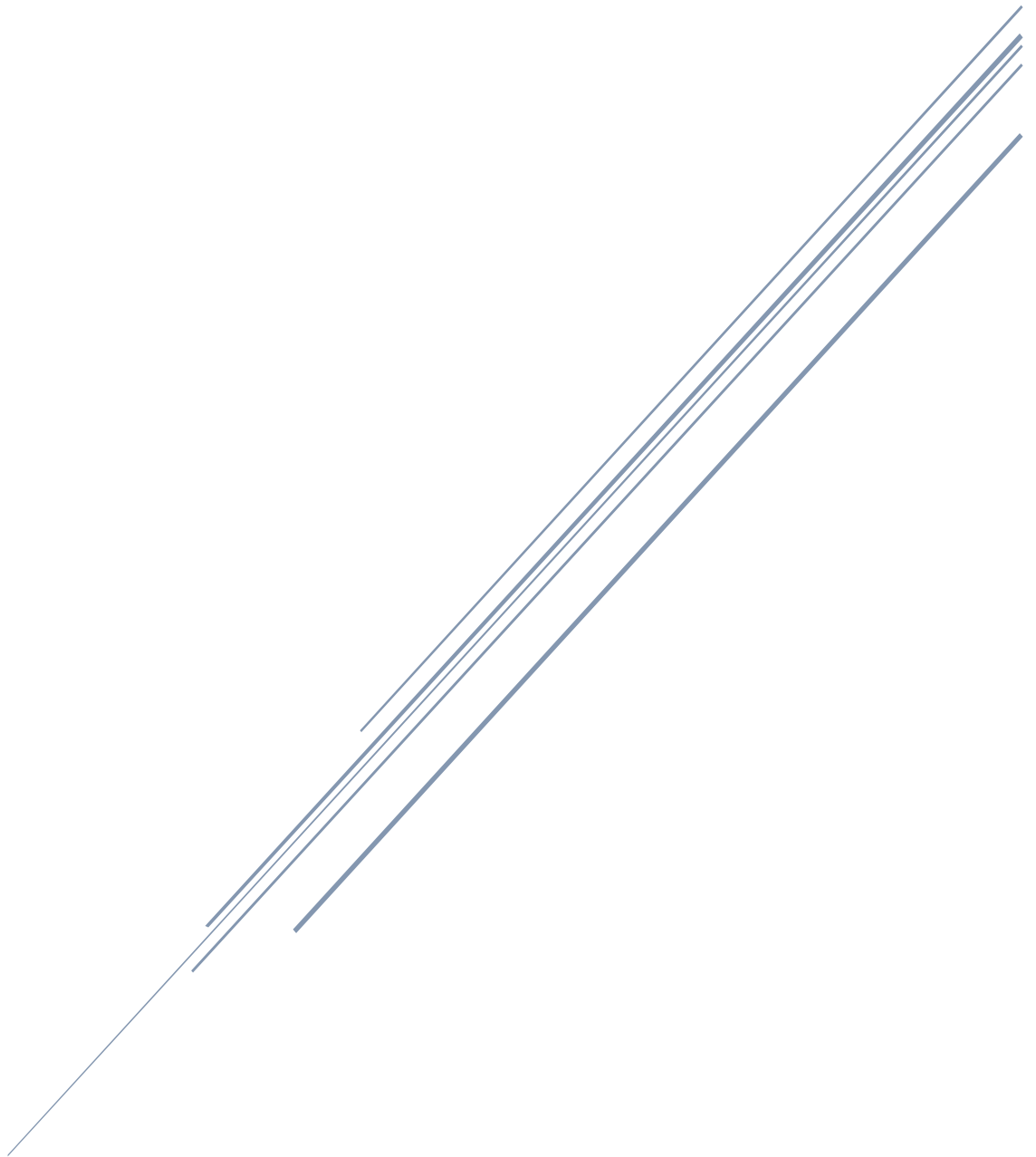


KÜNSTLICHE INTELLIGENZ DOKUMENTATION

Marc Pfitzenmaier, Björn Bauer (9973491)



Inhalt

1.	KONZEPT	2
A.1	ANNAHMEN FÜR DAS PROGRAMM	2
A.2	EINLESEN DER DATEN (LABYRINTH, START, ZIEL, FREIES TEIL)	2
A.3	LABYRINTH BAUTEILE	2
A.4	VERSCHIEBEN SPIELFIGUR (A*)	3
A.5	VERSCHIEBEN LABYRINTH (A*)	3
A.6	SPIELZUG	3
A.7	SPIELFIGUR NEU SETZEN	3
2.	BEZUG ZUR VORLESUNG	3
3.	BEGRÜNDUNG DES ENTWURFS/DER UMSETZUNG	4
4.	TEST UND ERGEBNISBEWERTUNG	5
5.	QUELLEN	5

1. Konzept

A.1 Annahmen für das Programm

- Der Spieler ist allwissend. Er sieht alle Spielsteine (das gesamte Labyrinth) und den Spielstein außerhalb.
- Das Bewegen der Figur ist optional, es kann demnach mehrmals hintereinander das Labyrinth verschoben werden.
- Die Position des Einschubs des nicht genutzten Spielsteins wird durch den Spieler bestimmt
 - o Der Spielstein kann sich nicht drehen
- Die Spielfigur wird, wenn sie rausfliegt auf das Feld gesetzt, dass sie hinausgeschoben hat (das Feld, das hineingeschoben wurde)

A.2 Einlesen der Daten (Labyrinth, Start, Ziel, freies Teil)

Bei Start des Programms müssen die oben genannten Daten eingelesen werden. Das Format dabei ist folgendes:

- Labyrinth
 - o Wird in Form einer CSV Datei eingelesen in der 5*4 Felder beschrieben sind (separiert durch ;).
- Start/Ziel
- Freies Teil
 - o Wird als Teil der Labyrinth-Datei eingelesen

A.3 Labyrinth Bauteile

Es handelt sich um ein 5*4 Felder großes Labyrinth, das aus 10 verschiedenen Bausteinen bestehen kann. Diese Bausteine werden, wie Bild 1 zeigt codiert

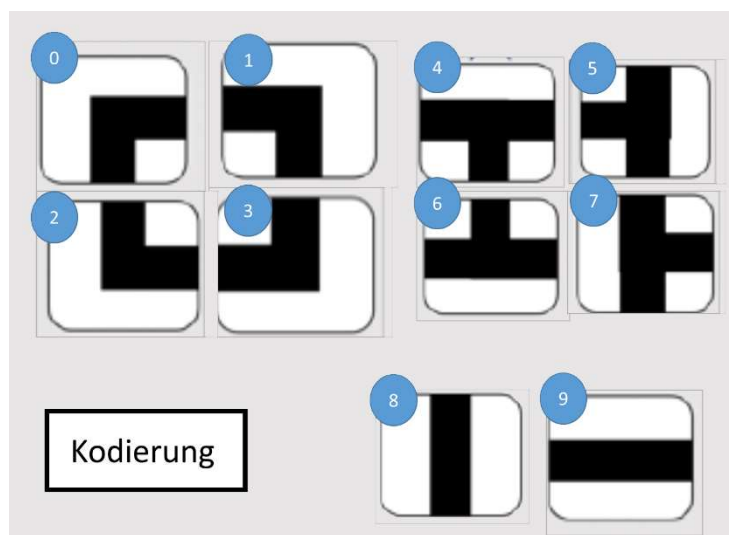


Abbildung 1: Codierung Labyrinth Bausteine

Jedes Labyrinth-Teil benötigt zusätzlich Regeln, in Welche Richtung gegangen werden kann und welches Bauteil an dieser Stelle möglich ist. Ein Beispiel hierfür wäre das bei Bauteil 0 nur nach rechts gegangen werden kann und auch nur, wenn sich hier das Bauteil 1, 4, 5, 6, 3 oder 9

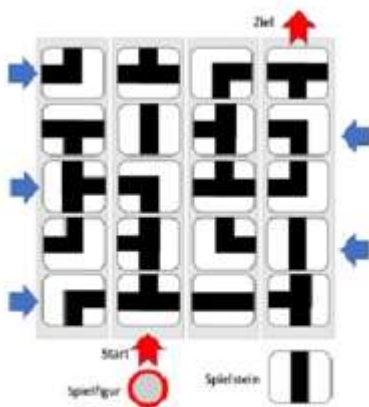
befindet. Außerdem kann nur nach unten gegangen werden, wenn sich hier Bauteil 2, 3, 8, 6, 5 oder 7 befindet. Ähnliche Regeln müssen für alle 10 Bauteile aufgestellt werden.

A.4 Verschieben Spielfigur (A*)

Die Spielfigur kann nur verschoben werden, wenn es einen Weg gibt auf dem Sie Laufen kann (in Schwarz). Dies betrifft auch Start und Ziel.

A.5 Verschieben Labyrinth (A*)

Das Labyrinth kann zu jeder Zeit verschoben werden, es ist nicht nötig, dass sich die Spielfigur zwischen dem Verschieben von Bauteilen bewegt. Spielsteine können nur an 5 spezifizierten Punkten eingesetzt werden (in Abbildung xxx durch blaue Pfeile markiert).



A.6 Spielzug

Ein Spielzug besteht aus dem Einsetzen eines Bauteils und dem anschließenden bewegen der Spielfigur. Die Spielfigur muss sich allerdings nicht bewegen weshalb jedes Einsetzen eines Spielsteins als einzelner Spielzug anzusehen ist.

A.7 Spielfigur neu setzen

Wird die Spielfigur aus dem Spielfeld geschoben (durch das Einsetzen eines Spielsteins) so wird Sie auf den Spielstein, welcher hineingeschoben wurde, wieder eingesetzt.

2. Bezug zur Vorlesung

Bei dieser Aufgabe handelt es sich um die Implementierung des A*-Algorithmus (siehe Abbildung xxx), ein Heuristisches Suchverfahren (Vorlesung Intelligente Suchverfahren) zur Bestimmung des kürzesten Wegs von einem Startpunkt zu einem Zielpunkt. Besonderheit bei

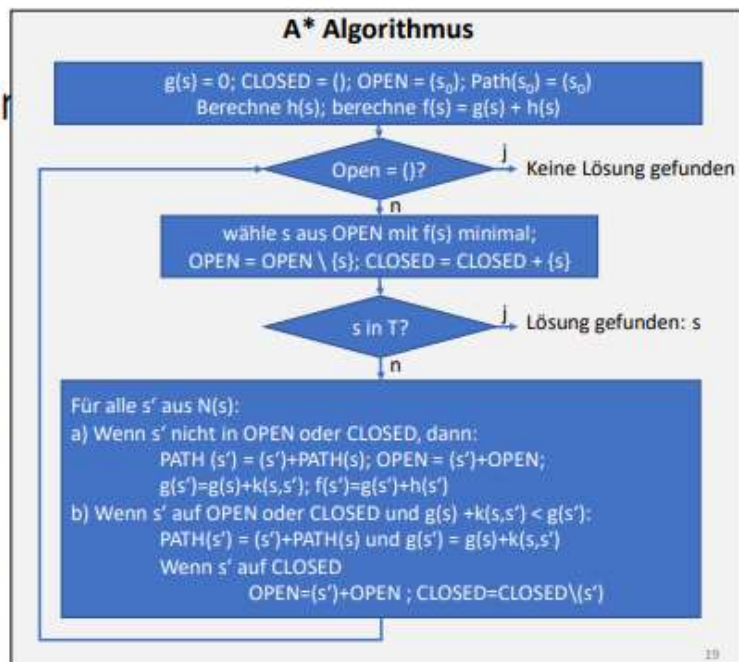
dieser Problemstellung ist, dass grundsätzlich zwei A* nötig sind, einen der den besten Weg für die Spielfigur findet und nach jedem Einsetzen des Spielsteins neu gestartet werden muss und einer, der den besten Punkt für das Einsetzen des Spielsteins bestimmt.

Suchstrat. - Pr

A* Algorithmus

Heuristikfunktion
 $f = g + h$
 = Kosten zur Erreichung des Knotens
 + Kosten vom Knoten zum Ziel

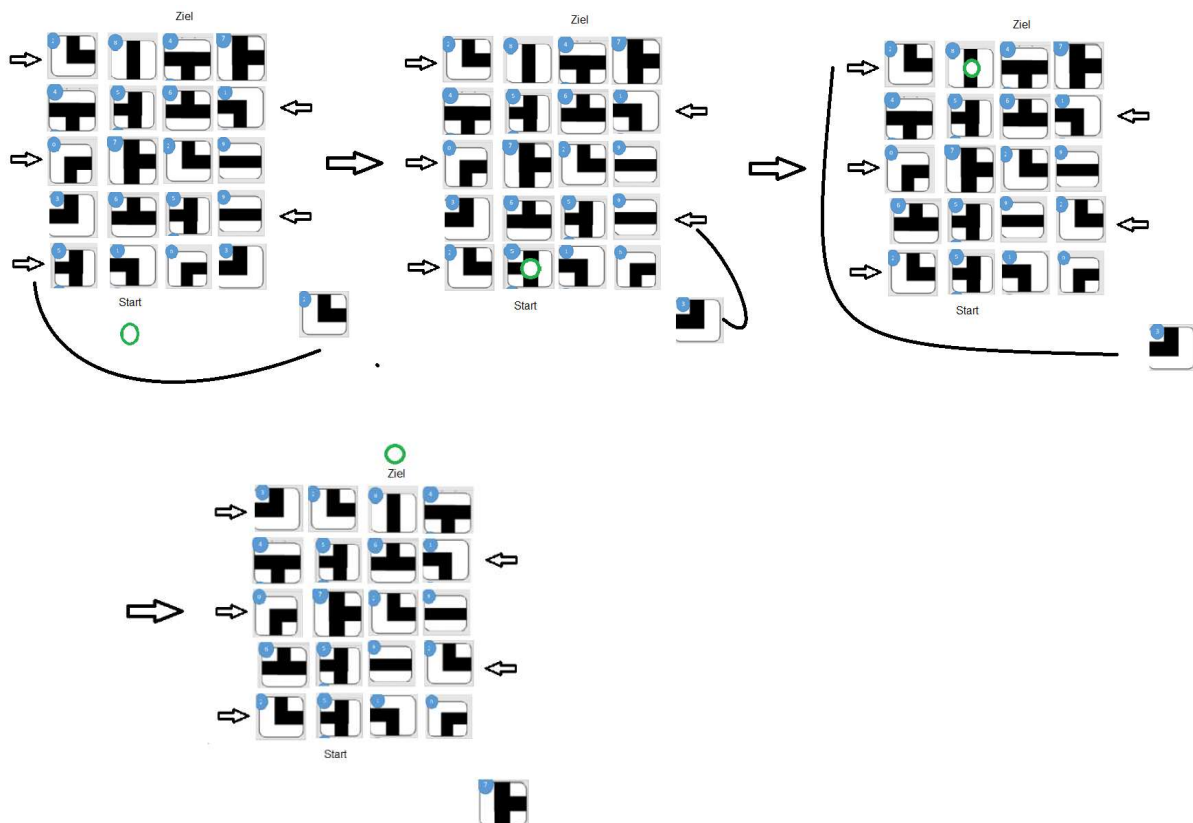
Open	Kosten $f = g + h$	Closed
S_1	3,8 + 16	
S_{21}, S_6	15 + 11,2; 9,4 + 14,4	S_1
S_{21}, S_{27}, S_9	15 + 11,2; 16,6 + 9,7; 16,8 + 10,4	S_{21}, S_6
S_{27}, S_{27}, S_9	26,4 + 2,5; 16,6 + 9,7; 16,8 + 10,4	S_{21}, S_{27}, S_6
S_{27}, S_{27}, S_9	26,4 + 2,5; 22,7 + 4,6; 16,8 + 10,4	S_{21}, S_{27}, S_6
S_{27}, S_{27}, S_{10}	26,4 + 2,5; 22,7 + 4,6; 22 + 10,4	S_{21}, S_{27}, S_6 S_{27}, S_9
S_{27}, S_{27}, S_{10}	26,4 + 2,5; 27,9 + 0; 22 + 10,4	S_{21}, S_{27}, S_6 S_{27}, S_9, S_{27}



3. Begründung des Entwurfs/der Umsetzung (Diskutieren der Konfiguration)

4. Test und Ergebnisbewertung (Diskutieren des Ergebnisses)

Das Labyrinth kann in 3 Zügen beendet werden (wie in Abbildung xxx gezeigt wird). Dabei muss vor jedem Laufen der Spielfigur ein Spielstein eingesetzt werden.



5. Quellen

Offene Punkte

- Lösungsqualität und Umfang der Funktionalität
- Korrekte Verwendung von Kernfunktionen
- Anpassung an die Aufgabenstellung